Fast exact digital differential analyzer for circle generation

Jan L. Cieśliński*

Uniwersytet w Białymstoku, Wydział Fizyki
ul. Lipowa 41, 15-424 Białystok, Poland

Leonid V. Moroz†

Lviv Polytechnic National University, Department of Security Information and Technology
st. Kn. Romana 1/3, 79000 Lviv, Ukraine

Abstract

In the first part of the paper we present a short review of applications of digital differential analyzers (DDA) to generation of circles showing that they can be treated as one-step numerical schemes. In the second part we present and discuss a novel fast algorithm based on a two-step numerical scheme (explicit midpoint rule). Although our algorithm is as cheap as the simplest one-step DDA algoritm (and can be represented in terms of shifts and additions), it generates circles with maximal accuracy, i.e., it is exact up to round-off errors.

Key words and phrases: circle generation, digital differential analyzers, exact discretization, explicit midpoint rule

1 Introduction

In spite of apparent simplicity fast generation of circles and other curves has always been a subject of numerous theoretical and practical studies, finding applications, among others, in digital plotting, graphical display and

*e-mail: janek@alpha.uwb.edu.pl
†e-mail: moroz_lv@polynet.lviv.ua

numerical machine tool control, compare [8, 11, 17, 21]. Leaving aside the fundamental algoritm of Bresenham [3] and techniques based on spline functions [18], in this paper we focus on digital differential analyzers (DDA), see for instance [7, 12, 15]. In section 2 we present brief but exhaustive discussion of digital differential analyzers from a unified point of view, treating them as special one-step difference numerical schemes. Then, in section 3, we introduce and discuss a new DDA scheme, cheap and accurate, based on a two-step numerical scheme.

In terms of the natural arc parameter ϑ the equation of the circle of radius r is $x = r\cos\vartheta$, $y = -r\sin\vartheta$. The corresponding differential equation is

$$\frac{dx}{d\vartheta} = -y \ , \quad \frac{dy}{d\vartheta} = x \ . \tag{1}$$

Considering various discretizations of (1) we can obtain any DDA algorithm.

2 One-step numerical approximations

In this section we present a unified approach to digital differential analysers representing them by one-step numerical schemes. Therefore they can be represented by a general matrix difference equation of the first order:

$$\left[\begin{array}{c} x_{n+1} \\ y_{n+1} \end{array}\right] = \left[\begin{array}{cc} a & b \\ c & d \end{array}\right] \left[\begin{array}{c} x_n \\ y_n \end{array}\right] , \qquad \left[\begin{array}{cc} a & b \\ c & d \end{array}\right] \equiv A \ , \tag{2}$$

where a, b, c, d depend on the time step h. In order to approximate the system (1) these coefficients have to satisfy

$$\begin{aligned} &\lim_{h\to 0} \left[\begin{array}{cc} a & b \\ c & d \end{array}\right] = \left[\begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array}\right] \equiv I, \\ &\lim_{h\to 0} \frac{A - I}{h} = \left[\begin{array}{cc} 0 & -1 \\ 1 & 0 \end{array}\right] \equiv J. \end{aligned} \tag{3}$$

In other words, $A = I + hJ + O(h^2)$, i.e.,

$$a = 1 + O(h^2), \quad d = 1 + O(h^2), \quad b = -h + O(h^2), \quad c = h + O(h^2).$$

The qualitative asymptotic behaviour of the discrete solution x_n, y_n is determined by the characteristic equation

$$0 = \det \left[\begin{array}{cc} a - \lambda & b \\ c & d - \lambda \end{array}\right] \equiv \lambda^2 - (a + d)\lambda + ad - bc \tag{4}$$

and its roots (eigenvalues of the matrix A) are given by

$$\begin{aligned} \lambda_1 &= \tfrac{1}{2}\left(a+d+\sqrt{(a+d)^2-4(ad-bc)}\right) = 1+ih+O(h^2), \\ \lambda_2 &= \tfrac{1}{2}\left(a+d-\sqrt{(a+d)^2-4(ad-bc)}\right) = 1-ih+O(h^2), \end{aligned} \tag{5}$$

where $i=\sqrt{-1}$.

2.1 Logarithmic spirals

The most popular DDA schemes have the form

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} a & -c \\ c & a \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} . \tag{6}$$

It is convenient to denote

$$\rho = \sqrt{a^2+c^2} \ , \qquad \theta = \arctan \frac{c}{a} \ , \tag{7}$$

i.e., $a=\rho\cos\theta$, $c=\rho\sin\theta$. Then,

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \begin{bmatrix} \rho\cos\theta & -\rho\sin\theta \\ \rho\sin\theta & \rho\cos\theta \end{bmatrix}^n \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} = \rho^n \begin{bmatrix} \cos n\theta & -\sin n\theta \\ \sin n\theta & \cos n\theta \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} .$$

Denoting $x_0 = r_0\cos\varphi_0$, $y_0 = r_0 \sin\varphi_0$, we finally get

$$x_n = r_0\rho^n \cos(n\theta+\varphi_0) \ , \quad y_n = r_0\rho^n\sin(n\theta+\varphi_0) \ . \tag{8}$$

Corollary 2.1. *All points generated by (6) lie on the logarithmic spiral defined by:*

$$r = r_0 e^{k(\varphi-\varphi_0)} \ , \qquad k := \theta^{-1}\ln\rho \ , \tag{9}$$

where r, φ are polar coordinates on the plane (x,y), compare [1].

From computational point of view the polynomial form of a, c is preferred. DDA algorithms can be classified by the order of these polynomials [12]. Below we present several schemes of this kind with the corresponding values of ρ^2 and k (the exact algorithm should have $\rho=1$ and $k=0$).

First order simultaneous DDA algorithm

$$a = 1 \ , \qquad c = h \ , \qquad \rho^2 = 1 + h^2 \ , \qquad k = \frac{h}{2} - \frac{h^3}{12} + \ldots \ . \tag{10}$$

This is the conventional DDA method [12].

Second order simultaneous DDA algorithm

$$a = 1 - \frac{1}{2}h^2 \ , \quad c = h \ , \quad \rho^2 = 1 + \frac{1}{4}h^4 \ , \quad k = \frac{h^3}{8} - \frac{h^5}{48} + \ldots \ , \tag{11}$$

see [2, 11].

Third order simultaneous DDA algorithm

$$a = 1 - \frac{1}{2}h^2, \quad c = h - \frac{1}{6}h^3, \quad \rho^2 = 1 - \frac{1}{12}h^4 + \frac{1}{36}h^6, \quad k = -\frac{h^3}{24} + \frac{h^5}{72} + \ldots, \tag{12}$$

compare [2].

Matsushiro's analyzer

$$a = 1 - \frac{1}{2}h^2, \quad c = h - \frac{1}{4}h^3, \quad \rho^2 = 1 - \frac{1}{4}h^4 + \frac{1}{16}h^6, \quad k = -\frac{h^3}{8} + \frac{h^5}{48} + \ldots. \tag{13}$$

An important point is that this scheme (with h replaced by $\frac{1}{2}h$) was implemented in terms of shifts and additions (without using multiplications) and patented, see [13, 14].

The best circular interpolator of the third order

Interestingly enough, the Matsushiro algoritm can be easily improved by changing the last term, see [4, 16]:

$$a = 1 - \frac{1}{2}h^2, \quad c = h - \frac{1}{8}h^3, \quad \rho^2 = 1 + \frac{1}{64}h^6, \quad k = \frac{h^5}{128} + \ldots. \tag{14}$$

We are going to show that this is the best spiral-like circular interpolator of the third order in h.

Lemma 2.2. *The most accurate circular interpolator of the form (6), where a, c are polynomials of the third order in h, is given by (14)*

Proof: We assume

$$a = 1 + a_1 h + a_2 h^2 + a_3 h^3 \ , \quad c = h + c_2 h^2 + c_3 h^3 \tag{15}$$

Then we compute $\rho^2 = a^2 + c^2$:

$$\rho^2 = 1 + 2a_1 h + (2a_2 + a_1^2 + 1)h^2 + 2(a_3 + a_1 a_2 + c_2)h^3 + \ldots + (a_3^2 + c_3^2)h^6 .$$

Equating to zero five first coefficients we obtain a system of 5 equations for 5 unknowns, which can be solved easily:

$$\begin{aligned}
&2a_1 = 0 \quad \Longrightarrow \quad a_1 = 0 \ , \\
&2a_2 + a_1^2 + 1 = 0 \quad \Longrightarrow \quad a_2 = -\tfrac{1}{2} \ , \\
&a_3 + a_1 a_2 + c_2 = 0 \quad \Longrightarrow \quad a_3 = -c_2 \ , \\
&a_2^2 + 2a_1 a_3 + c_2^2 + 2c_3 = 0 \quad \Longrightarrow \quad c_3 = -\tfrac{1}{8} - \tfrac{1}{2}c_2^2 \ , \\
&a_2 a_3 + c_2 c_3 = 0 \quad \Longrightarrow \quad \tfrac{1}{2}c_2 \left(\tfrac{3}{4} - c_2^2\right) = 0 \ .
\end{aligned} \tag{16}$$

Then, the coefficient by h^6 is given by $a_3^2 + c_3^2 = c_2^2 + \frac{1}{4}\left(\frac{1}{4} + c_2^2\right)^2$. The last equation of (16) yields two possibilities: either $c_2 = 0$ or $c_2^2 = \frac{3}{4}$. Corresponding coefficients by h^6 are given by $\frac{1}{64}$ and 1, respectively. Therefore the best accuracy is attained in the first case, which leads to (14) (the second case has another disadvantage: coefficients c_2 and a_3 are irrational).

Interestingly enough, considering h-expansion of the function $k = k(h)$ we get the same final results (although the starting system of equations is more complicated). The case $c_2 = 0$ yields $k = \frac{1}{128}h^5 + O(h^6)$ while in the second case, $c_2^2 = \frac{3}{4}$, we get $k = \frac{1}{2}h^5 + O(h^6)$. □

2.2 Elliptical deformations

Discrete points generated by (2) lie on an ellipse if and only if $\bar{\lambda}_2 = \lambda_1$ and $|\lambda_1| = 1$, which is equivalent to

$$|a + d| < 2 \quad \text{and} \quad ad - bc = 1 \ . \tag{17}$$

The conditions are known as Barkhausen criteria [19].

First order sequential DDA algorithm

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & -h \\ h & 1-h^2 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} . \tag{18}$$

This is the generator of the so called "magic circle" [13, 17, 19].

2.3 Elliptical spirals

Two complex eigenvalues ($\lambda_2 = \bar{\lambda}_1$) are obtained when

$$(a-d)^2 + 4bc < 0 \ , \tag{19}$$

and, for $ad - bc \neq 1$ the generated circle deforms into an elliptical spiral. The following example can be found in [12].

Second order sequential DDA algorithm

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1-\frac{1}{2}h^2 & -h \\ h & 1-\frac{3}{2}h^2 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \tag{20}$$

One can easily verify, that $(a-d)^2 + 4bc = h^2(h^2-4)$. Therefore (19) is satisfied for $|h| < 2$.

Sequentials algorithms are derived from simulataneous algorithms by replacing x_n by x_{n+1} in the equation for y_{n+1}, i.e., simultaneous algorithm (6) of order M is transformed into

$$\begin{aligned} x_{n+1} &= ax_n - cy_n \ , \\ y_{n+1} &= cx_{n+1} + ay_n = acx_n + (a-c^2)y_n \ , \end{aligned} \tag{21}$$

and then truncated up to the order M, see [12]. In the case $M = 2$ we get the scheme (20).

2.4 Exactly circular interpolation

Digital differential analyzer of the form (2) yields an exact circle (up to round-off errors) if and only if $a = d$, $c = -b$ and $ad - bc = 1$. Then, the matrix A can be parameterized by a single parameter θ

$$A = \begin{bmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{bmatrix} , \tag{22}$$

where $\phi = \phi(h)$ depends on time step h. In other words,

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} C(h) & -S(h) \\ S(h) & C(h) \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \tag{23}$$

where $C^2(h) + S^2(h) = 1$. Taking $\theta(h) = h$ we obtain the exact numerical scheme (compare [17, 20])

$$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} \cos h & -\sin h \\ \sin h & \cos h \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} . \tag{24}$$

Exact numerical schemes can be constructed for any matrix differential equation of the first order, see [5, 6].

The Taylor expansion of (24) up to the third order yields the scheme (12), compare [2]. We recall that this is not the best scheme of the third order, see Lemma 2.2.

Implicit midpoint rule

In the case of linear equations the implicit midpoint rule can be represented in an explicit form. Indeed,

$$\frac{x_{n+1} - x_n}{h} = -\frac{y_n + y_{n+1}}{2} \ , \qquad \frac{y_{n+1} - y_n}{h} = -\frac{x_n + x_{n+1}}{2} \ , \tag{25}$$

can be rewritten in a matrix form as follows

$$\begin{bmatrix} 1 & \frac{1}{2}h \\ -\frac{1}{2}h & 1 \end{bmatrix} \begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1}{2}h \\ \frac{1}{2}h & 1 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} . \tag{26}$$

Thus we obtain another algortihm of the form (6), where

$$a = \frac{4 - h^2}{4 + h^2} \ , \qquad c = \frac{4h}{4 + h^2} \ . \tag{27}$$

see [11]. Note that $a^2 + c^2 = 1$ and $k = 0$, so the implicit midpoint rule generates the circle exactly. However, this algorithm has a relatively high cost (multiplications in every step). Actually the scheme (27) was a motivation for the derivation of the Matsushiro algorithm (13), see [13]. Indeed, (13) is the Taylor truncation of (27) up to the third order.

3 New fast circular interpolator

The main result of our paper consists in applying a two step method to the circular interpolation. The explicit midpoint rule (see [10]), known also as 2-step Nyström method [9], applied to a general ordinary differential equation $\dot{\boldsymbol{z}} = \boldsymbol{f}(t, \boldsymbol{z})$ (where $\boldsymbol{z} \in \mathbb{R}^n$ and $\boldsymbol{f}$ is a given function) reads:

$$\boldsymbol{z}_{n+2} = \boldsymbol{z}_n + 2h\boldsymbol{f}(t_{n+1}, \boldsymbol{z}_{n+1}) \ . \tag{28}$$

In the case of the circular interpolation $\boldsymbol{z} = \begin{bmatrix} x \\ y \end{bmatrix}$ and $\boldsymbol{f}(t, \boldsymbol{z}) \equiv \begin{bmatrix} -y \\ x \end{bmatrix}$. Therefore we obtain:

$$\begin{aligned} x_{n+2} &= x_n - 2hy_{n+1} \ , \\ y_{n+2} &= y_n + 2hx_{n+1} \ , \end{aligned} \tag{29}$$

compare [16]. This is a two step method and we have to prescribe not only x_0, y_0 but also x_1, y_1. As usual, we can rewrite this scheme in a one-step form increasing the dimension of the matrix problem, namely:

$$\begin{bmatrix} x_{n+2} \\ y_{n+2} \\ x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 0 & -2h & 1 & 0 \\ 2h & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_{n+1} \\ y_{n+1} \\ x_n \\ y_n \end{bmatrix} . \tag{30}$$

3.1 First integrals

In this section we are going to show that (29) preserves circular trajectories exactly provided that initial conditions are appropriately chosen.

Lemma 3.1. *If $x_n^2 + y_n^2$ is a first integral of the system (29), then $x_n y_{n+1} - x_{n+1} y_n$ is also a first integral. What is more, in this case both first integrals are linearly dependent, namely*

$$x_n y_{n+1} - x_{n+1} y_n = h(x_n^2 + y_n^2) \tag{31}$$

Proof: Using (29) we compute

$$x_{n+2}^2 + y_{n+2}^2 = x_n^2 + y_n^2 - 4h(x_n y_{n+1} - x_{n+1} y_n) + 4h^2(x_{n+1}^2 + y_{n+1}^2) \ .$$

If $x_n^2 + y_n^2 = r^2 = \text{const}$, then, obviously, $x_n y_{n+1} - x_{n+1} y_n = hr^2$. □

Motivated by above results, we define:

$$\boldsymbol{\xi}_n = \begin{bmatrix} x_n^2 + y_n^2 \\ x_{n+1}^2 + y_{n+1}^2 \\ x_n y_{n+1} - x_{n+1} y_n \end{bmatrix} , \quad B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4h^2 & -4h \\ 0 & 2h & -1 \end{bmatrix} . \tag{32}$$

Lemma 3.2. *For any x_n, y_n satisfying (29) we have*

$$\boldsymbol{\xi}_{n+1} = B\boldsymbol{\xi}_n , \tag{33}$$

Proof: A simple straightforward calculation. □

Theorem 3.3. *All points generated by (29) lie exactly (up to round-off errors) on the circle $x + y = r^2$ provided that initial conditions satisfy*

$$x_0^2 + y_0^2 = r^2 , \qquad x_1^2 + y_1^2 = r^2 , \qquad x_0 y_1 - x_1 y_0 = hr^2 . \tag{34}$$

Proof: We easily verify that

$$B f_1 = f_1 \ , \qquad \boldsymbol{f}_1 := \begin{bmatrix} 1 \\ 1 \\ h \end{bmatrix} . \tag{35}$$

Initial conditions (46) can be shortly written as $\boldsymbol{\xi}_0 = r^2 \boldsymbol{f}_1$. By (33) we have $\boldsymbol{\xi}_n = \boldsymbol{\xi}_0$ for any n. It means that $x_n^2 + y_n^2 = r^2$ for any n. □

Remark 3.4. *The simplest initial conditions satisfying (46) read*

$$x_0 = r \ , \quad y_0 = 0 \ , \quad x_1 = r\sqrt{1 - h^2} \ , \quad y_1 = hr \ , \tag{36}$$

compare [16].

3.2 Low computational cost

The scheme (29) has a very low computation cost, to be compared only with the simplest one-step scheme (10). Indeed, both (29) and (10) can be implemented in terms of two additions and two shifts. However, (29) produces an exact circle while the accuracy of (10) is rather limited.

There is only one more complicated computation, at the start, when initial conditions are x_1, y_1 are determined. However, taking into account the Taylor expansion

$$\sqrt{1-h^2} = 1 - \frac{1}{2}h^2 - \frac{1}{8}h^4 - \frac{1}{16}h^6 - \frac{5}{128}h^8 - \frac{7}{256}h^{10} + O(h^{12}) \qquad (37)$$

we see that even in that case one can use few shifts to produce x_1. Indeed, let $h = 2^{-m}$ and $r = 2^N$. Then

$$\sqrt{1-h^2} = 1 - 2^{-2m-1} - 2^{-4m-3} - 2^{-6m-4} - 2^{-8m-5} - 2^{-8m-7} + \ldots \quad (38)$$

and

$$x_1 = 2^N - 2^{N-2m-1} - 2^{N-4m-3} - 2^{N-6m-4} - 2^{N-8m-5} - 2^{N-8m-7} + \ldots \quad (39)$$

where only integer coefficients are left.

The case $h = \frac{1}{2}$ (i.e., $m = 1$) corresponds to the circle approximated by regular dodecagon (because then $x_1 = r\cos\frac{\pi}{6}$, $y_1 = r\sin\frac{\pi}{6}$, compare (36)). Assuming $N = 8$ we may confine ourselves to the first three terms.

3.3 Stability

Characteristic equation $\det(B - \lambda I) = 0$ reads

$$(1-\lambda)(\lambda^2 + 2(1-2h^2)\lambda + 1) = 0, \qquad (40)$$

hence eigenvalues of B are given by:

$$\lambda_1 = 1, \quad \lambda_2 = 2h^2 - 1 + 2h\sqrt{h^2-1}, \quad \lambda_3 = 2h^2 - 1 - 2h\sqrt{h^2-1}. \quad (41)$$

All the eigenvalues lie on the unit circle: $|\lambda_k| = 1$ for $k = 1, 2, 3$. Corresponding eigenvectors are denoted by $\boldsymbol{f}_k$, $k = 1, 2, 3$, namely

$$B\boldsymbol{f}_1 = f_1 \ , \quad B\boldsymbol{f}_2 = \lambda_2\boldsymbol{f}_2 \ , \quad B\boldsymbol{f}_3 = \lambda_3\boldsymbol{f}_3, \qquad (42)$$

($\boldsymbol{f}_1$ is given by (35) and the form of other eigenvectors is not important). The initial condition $\boldsymbol{\xi}_0$ can always be represented as a linear combination of eigenvectors: $\boldsymbol{\xi}_0 = a\boldsymbol{f}_1 + b\boldsymbol{f}_2 + c\boldsymbol{f}_3$, where we assume $a \approx r^2$ (compare Theorem 3.3) and b, c are small perturbations. Using (42) and (33) we obtain

$$\boldsymbol{\xi}_n = \boldsymbol{f}_1 + b\lambda_2^n\boldsymbol{f}_2 + c\lambda_3^n\boldsymbol{f}_3. \qquad (43)$$

Taking into account $|\lambda_2^n| = |\lambda_3^n| = 1$, we see that the initial perturbations remain unchanged during the evolution.

3.4 Exact circular interpolator

The scheme (29) produces exact circular trajectory but the period is modified. For instance, in the case of regular dodecagon ($h = \frac{1}{2}$) the total change of ϑ after 12 steps is $\Delta\vartheta = 12h = 6$ while the exact value is, of course, 2π. In general, the period of the circular interpolator (29) is given by

$$T = \frac{2\pi h}{\arcsin h} . \tag{44}$$

In order to preserve exactly not only the trajectory but also the period of the circular motion, we can use a nonstandard modification of (29), following the approach of [5, 6]:

$$\begin{aligned} x_{n+2} &= x_n - 2\delta y_{n+1} \ , \\ y_{n+2} &= y_n + 2\delta x_{n+1} \ , \end{aligned} \tag{45}$$

where $\delta = \delta(h)$ is a given function of h.

Theorem 3.5. *For any $\delta = \delta(h)$ all points generated by (45) lie exactly (up to round-off errors) on the circle $x + y = r^2$ provided that initial conditions satisfy*

$$x_0^2 + y_0^2 = r^2, \qquad x_1^2 + y_1^2 = r^2, \qquad x_0 y_1 - x_1 y_0 = \delta r^2 . \tag{46}$$

Proof: It is enough to repeat the proof of Theorem 3.3 (with h replaced by δ in appropriate places). □

The period of the circular interpolator (45) is

$$T = \frac{2\pi h}{\arcsin \delta(h)} . \tag{47}$$

Theorem 3.6. *If $\delta = \sin h$, then scheme*

$$\begin{aligned} x_{n+2} &= x_n - 2 y_{n+1} \sin h \ , \\ y_{n+2} &= y_n + 2 x_{n+1} \sin h \ , \\ x_0 y_1 - x_1 y_0 &= r^2 \sin h \ , \end{aligned} \tag{48}$$

generates the circle exactly (up to round-off errors) preserving the period.

Proof: The scheme (45) will preserve the period exactly if $x_n = r\cos hn$ and $y_n = r\sin hn$. Taking into account well known trigonometric identities

$$\begin{aligned} \cos(hn+2h) - \cos(hn) &= -2\sin h \sin(hn+h) \ , \\ \sin(hn+2h) - \sin(hn) &= 2\sin h \cos(hn+h) \ , \end{aligned} \tag{49}$$

we conclude that scheme (45) is exact for $\delta(h) = \sin h$. □

The exact interpolator (48) uses two multiplication by $\sin h$ at every step, hence it is relatively expensive (but two times cheaper than its one-step conterpart (24)).

The scheme (45) can serve as a starting point for deriving cheaper algorithms by taking h-polynomials instead of $\sin h$. All these schemes preserve exactly the circular trajectory. The best approximation of the period are obtained for Taylor truncations. In particular, taking

$$\delta = h - \frac{1}{6}h^3 \tag{50}$$

we get the best scheme (45) of order 3. The multiplication by $\frac{1}{6}$ can be replaced by shifts because

$$\frac{1}{6} = \frac{1}{8\left(1-\frac{1}{4}\right)} = \frac{1}{8}\sum_{k=0}^{\infty} 4^{-k} = \sum_{k=0}^{\infty} 2^{-3-2k}, \tag{51}$$

and we can approximate δ by δ_N given by

$$\delta_N = h - h^3 \sum_{k=0}^{N} 2^{-3-2k}, \tag{52}$$

where N is chosen to assure the required accuracy.

4 Conclusions

We discussed DDA circular interpolators based on one-step numerical schemes. We have shown that the best cheap interpolator of the third order is given by (14), see Lemma 2.2. This algorithm can be expressed in terms of shifts and additions only, and it has quite good accuracy. Exact interpolators based on one-step methods are more expensive because multiplications in every step are required.

In section 3 we presented a family of circular interpolators, based on a two-step method, namely the explicit midpoint rule. All of them interpolate the circle exactly (up to round-off errors). The simplest interpolator, given by (29), is very cheap and is very good for graphical applications (when the period of the circular motion is irrelevant). The scheme (48) preserves exactly also the period but is more expensive. Taking an appropriate polynomial $\delta(h)$ we can control the accuracy and computational cost of the scheme (45), obtaining interpolators suitable for different purposes.

Acknowledgement. The first author (J.L.C.) is partly supported by the National Science Centre (NCN) grant no. 2011/01/B/ST1/05137.

References

[1] C.Baumgarten, G.Farin: Approximation of logarithmic spirals, *Computer Aided Geometric Design* **14** (6) (1997) 515-532.

[2] C.A.Bergren: A simple algorithm for circular interpolation, *Control Engineering* **18** (9) (1971) 57-59.

[3] J.E.Bresenham: A linear algorithm for incremental digital display of circular arcs, *Commun. ACM* **20** (2) (1977) 100-106.

[4] I.Chami: A high precision digital differential analyzer for circular interpolation, *Tishreen Univ. J. Stud. Sci. Res., Eng. Sci. Ser.* **29** (1) (2007).

[5] J.L.Cieśliński, B.Ratkiewicz: On simulations of the classical harmonic oscillator equation by difference equations, *Adv. Difference Equ.* **2006** (2006) 40171 (17pp).

[6] J.L.Cieśliński: On the exact discretization of the classical harmonic oscillator equation, *J. Difference Equ. Appl.* **17** (2011) 1673-1694.

[7] P.E.Danielsson: Incremental curve generation, *IEEE Trans. Comput.* **C-19** (9) (1970) 783-793.

[8] X.Y.Fan, Y.H.Guo, S.C.Li: New digital differential analyzer interpolation algorithm, *Advanced Science Letters* **6** (2012) 692-695.

[9] E.Hairer, S.P.Nørsett, G.Wanner: *Solving ordinary differential equations I. Nonstiff problems*, Springer, Berlin 1987.

[10] A.Iserles: *A first course in the numerical analysis of differential equations*, second edition, Cambridge Univ. Press 2009.

[11] Y.Koren: Interpolator for a computer numerical control system, *IEEE Trans. Comput.* **C-25** (1) (1976) 32-37.

[12] F.S.Lim, Y.S.Wong, M.Rahman: Circular interpolators for numerical control: A comparison of the modified DDA techniques and an LSI interpolator, *Computers in Industry* **18** (1992) 41-52.

[13] N.Matsushiro: A new digital differential analyzer for circle generation, *IEICE Trans. Inf. Syst.* **E81-D** (2) (1998) 239-242.

[14] N.Matsushiro, I.Oyake: Method of and device for circle generation, *US Patent* No. 4,999,797 (March 1991).

[15] P.G.McCrea, P.W.Baker: On digital differential analyzer (DDA) circle generation for computer graphics, *IEEE Trans. Comput.* **C-24** (11) (1975) 1109-1110.

[16] L.V.Moroz: A method for studying characteristics of digital sine-cosine generators, *Information Extraction and Processing* 36 (112) (2012) 84-90 [in Ukrainian].

[17] W.M.Newman, R.F.Sproull: *Principles of interactive computer graphics*, McGraw-Hill 1979.

[18] L.A.Piegl, W.Tiller: Circle approximation using integral B-splines, *Computer-Aided Design* **35** (2003) 601-607.

[19] C.S.Turner: Recursive discrete-time sinusoidal oscillators, *IEEE Signal Processing Magazine*, May 2003.

[20] W.P.Wang, C.Y.Wang: Difference method for generation of circular arcs and ellipses, *Computer-Aided Design* **21** (1) (1989) 33-37.

[21] L.Yong-Kui: Algorithm for circle approximation and generation, *Computer-Aided Design* **25** (3) (1993) 169-171.